

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
„ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних занять з дисципліни
“ЕКСПЕРТНІ СИСТЕМИ ТА БАЗИ ЗНАНЬ”
для студентів спеціальностей
7.080201 “Інформатика”,
7.080202 “Прикладна математика”,
7.080203 “Системний аналіз і управління”

Затверджено
редакційно-видавничою
радою університету,
протокол № 3 від 03.12.2008

Харків НТУ “ХПІ” 2009

Методичні вказівки до лабораторних занять з дисципліни “Експертні системи та бази знань” для студентів спеціальностей 7.080202 “Прикладна математика”, 7.080203 “Системний аналіз і управління” та 7.080201 “Інформатика” / уклад. Ю.І. Дорофєєв. – Харків: НТУ “ХПІ”, 2009. – 28 с.

Укладач Ю.І. Дорофєєв

Рецензент В.П. Северин

Кафедра системного аналізу і управління

ВСТУП

Експертні системи були розроблені як науково-дослідні інструментальні засоби в 1960-х роках ХХ століття і розглядалися як штучний інтелект спеціального типу, призначений для розв'язання складних задач у вузькій предметній області. Комерційне впровадження експертних систем відбулося на початку 1980-х років, і відтоді експертні системи набули широкого поширення. У цей час експертні системи успішно використовуються в бізнесі, науці, техніці, на виробництві, а також у багатьох інших сферах, де сформувалася цілком певна предметна область.

Перелік типових задач, для розв'язання яких доцільно застосовувати експертні системи, включає:

- інтерпретацію даних (таких, як сигнали, що надходять від гідролокатора);
- діагностику несправностей (як у технічних системах, так і в людському організмі);
- структурний аналіз складних об'єктів (наприклад, хімічних сполук);
- вибір конфігурації багатокомпонентних систем (наприклад, розподілених комп'ютерних систем);
- планування послідовності виконання операцій, що приводять до заданої мети (наприклад, виконуваних промисловими роботами);
- прогнозування наслідків подій або явищ на підставі даних про їх поточний стан (наприклад, прогнозування врожаю залежно від поширення шкідників).

Метою цих методичних вказівок є допомога студентам у засвоєнні на практиці основних методів здобуття і обробки предметних знань для побудови бази знань продукційного типу, а також принципів побудови та функціонування експертних систем.

У результаті виконання лабораторних робіт студенти повинні вивчити принципи функціонування механізмів логічного виведення різного типу, способів їхньої реалізації та тестування. Також студенти повинні ознайомитися з принципами роботи редактора бази знань та підсистеми пояснення експертної системи.

Оскільки зазначена галузь досліджень безперервно розвивається, даний посібник описує лише основні типи архітектури експертних систем та структури даних, які є необхідними для роботи механізму логічного виведення.

Дані лабораторні роботи спрямовані на набуття практичних навичок використання сучасних програмних засобів для побудови баз знань та їх застосування в експертних системах продукційного типу.

Лабораторна робота 1

**Розробка елементів експертної системи продукційного типу,
яка реалізує пряме виведення**

1.1. Мета роботи

Вивчити склад і принципи роботи експертної системи продукційного типу. Реалізувати механізм логічного виведення експертної системи, яка реалізує пряме виведення.

1.2. Інформаційний матеріал

1.2.1. Склад і принципи функціонування експертної системи

До складу експертної системи обов'язково входять підсистеми, які зображені на рис. 1.1:

- база знань;
- робоча пам'ять;
- механізм логічного виведення.

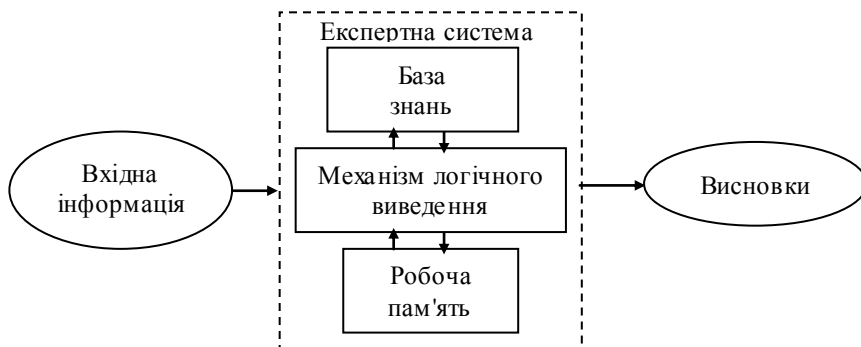


Рисунок 1.1 – Склад експертної системи

Робоча пам'ять містить набір фактів, які описують поточну ситуацію. База знань забезпечує зберігання знань, які подані за допомогою однієї з моделей: логічної, фреймової, мережної, продукційної [1]. Механізм логічного виведення, використовуючи дані й знання, організує керування висновком. Процедури, які реалізують цикл керування, є відділеними від експертних знань про те, як розв'язуються задачі в обраній предметній області.

1.2.2. Продукційна модель зображення знань

Продукційна модель зображення знань є однією з найпоширеніших. Знання в такій моделі подаються за допомогою правил вигляду:

«ЯКЩО $\{A\}$, ТО $\{B\}$ »,

де A – послідовність виражень (зразків), об'єднаних за допомогою логічних операцій «І», «АБО»; B – послідовність дій, які можуть модифікувати стан робочої пам'яті.

Продукційні системи були запропоновані в 1943 році американським математиком Е. Постом і розглядалися як модель організації обчислювального процесу. Подальший розвиток вони одержали в роботах А. Ньюела і Г. Саймона, які встановили, що продукційні правила відповідають елементам знань, що накопичуються у довгостроковій пам'яті людини. Такі елементи знань активізуються, якщо виникає відповідна ситуація. Робоча пам'ять продукційних систем аналогічна короточасній пам'яті людини, яка утримує в центрі уваги поточну ситуацію. Вміст робочої пам'яті, як правило, не зберігається після розв'язання задачі. Продукційні моделі зображення знань набули поширення в експертних системах, які моделюють розв'язання задач людиною-експертом у вузькій предметній області.

У порівнянні з іншими моделями зображення знань продукційні моделі мають наступні переваги:

- універсальність – практично будь-яка галузь знань може бути зображена в продукційній формі;
- модульність – кожна продукція являє собою відокремлений фрагмент знань про деяку проблему з обраної предметної області, видалення одних і додавання інших продукцій виконується незалежно;
- декларативність – продукції визначають ситуації предметної області, а не механізм керування;
- природність процесу виведення висновків, який багато в чому аналогічний процесу міркувань експерта.

Як приклад, розглянемо правило, яке взяте з бази знань експертної системи MYCIN, призначеної для діагностики інфекційних захворювань:

«ЯКЩО місце виділення культури – кров

І реакція мікроорганізму – грамнегативна

І форма мікроорганізму – паличка,

ТО назва мікроорганізму – *pseudomonias aeruginosa*».

Умовна частина правила складається з чотирьох фактів, які з'єднані за допомогою сполучника «і». У базі знань факти зручно подавати за допомогою триплету: *об'єкт-атрибут-значення*. Так, розглянуте правило можна подати у вигляді таблиці 1.1.

Таблиця 1.1

| | Об'єкт | Атрибут | Значення |
|----------|---------------|---------|-------------------------|
| Умова | культура | місце | кров |
| | мікроорганізм | реакція | грамнегативна |
| | мікроорганізм | форма | паличка |
| Висновок | мікроорганізм | назва | pseudomonias aeruginosa |

Якщо в процесі активізації правила три факти, що становлять умовну частину, виявляться істинними, то в робочу пам'ять буде поміщений новий факт, який подає висновок правила.

1.2.3. Механізм логічного виведення

Процес логічного виведення є циклічним і називається *пошуком за зразком*. Зразок – це деяка інформаційна структура, що визначає узагальнену ситуацію (умову, стан і т.п.), за якої активізується правило. Робоча пам'ять відбиває конкретні ситуації, що виникають у зовнішньому середовищі. Інформаційна структура, що подає конкретну ситуацію зовнішнього середовища в робочій пам'яті, називається *образом*.

Розглянемо процес логічного виведення в спрощеній формі. Поточний стан предметної області, яка моделюється, відбивається в робочій пам'яті у вигляді сукупності образів, кожен з яких подається за допомогою фактів. Робоча пам'ять ініціюється фактами, які описують вихідні дані задачі. Потім з бази знань вибираються ті правила, для яких зразки, що подаються умовними частинами правил, збігаються з образами в робочій пам'яті. Відібрані правила утворюють конфліктну множину. Всі правила, що входять у конфліктну множину, можуть бути активовані в поточному циклі, але відповідно до обраного механізму розв'язання конфлікту активується лише одне з правил. Виконання дії, яка міститься у висновку правила, приводить до зміни стану робочої пам'яті. Надалі цикл керування виведенням повторюється. Зазначений процес завершується, коли в базі знань не виявиться правил, передумови яких збігаються з образами робочої пам'яті.

1.2.4. Приклад побудови бази знань продукційного типу

Розглянемо предметну область, яка стосується рибного лову. Припустимо, що умовна частина продукційного правила не може містити більше двох висловлень, об'єднаних за допомогою логічної операції кон'юнкції.

Побудуємо дерево логічних висновків, що відбиває процес прийняття рішень в обраній предметній області (рис. 1.2).

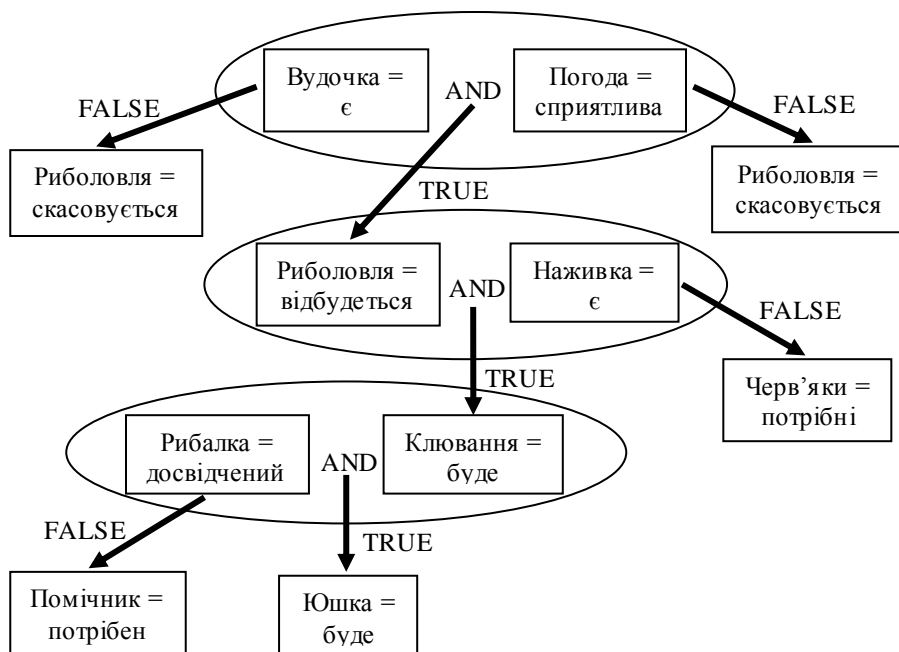


Рисунок 1.2 – Дерево логічних висновків

Кожна стрілка позначає логічний висновок, якому в базі знань відповідає одне продукційне правило. Таким чином, для наведеного прикладу побудуємо базу знань, яка містить сім правил:

1. ЯКЩО «вудочка = є»

І «погода = сприятлива», ТО «риболовля = відбудеться».

2. ЯКЩО «вудочка = НЕ є», ТО «риболовля = скасовується».

3. ЯКЩО «погода = НЕсприятлива», ТО «риболовля = скасовується».

4. ЯКЩО «риболовля = відбудеться»

I «наживка = є»,

ТО «клювання = буде».

5. ЯКЩО «наживка = НЕ є»,

ТО «черв'яки = потрібні».

6. ЯКЩО «клювання = буде»

I «рибалка = досвідчений»,

ТО «юшка = буде».

7. ЯКЩО «рибалка = НЕ досвідчений», ТО «помічник = потрібний».

Після формування бази знань створюються інші структури даних:

1) список змінних умови, тобто перелік змінних, які увійшли до умовних частин кожного правила. Являє собою масив, у якому для кожного правила резервується деяка кількість елементів, у даному прикладі – два (табл. 1.2);

2) робоча пам'ять, яка містить імена змінних, ознаку їхньої ініціалізації та присвоєні значення (табл. 1.4). До початку експертизи ознака ініціалізації дорівнює NI (англ.: No Initialize) і всім змінним присвоєні порожні значення. У процесі експертизи змінним присвоюються поточні значення, а значення ознаки змінюється на I (англ.: Initialize);

3) показчик змінних умови, який виконує роль протоколу роботи системи. Являє собою масив, у який заноситься номер поточного правила, номер умови правила, що перевіряється в цей момент, і результат перевірки істинності (табл. 1.3);

4) черга змінних логічного виведення, у якій прийнятий такий принцип обробки: «першим прийшов – першого обслужили».

Таблиця 1.2

| № з/п | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|--------------|-------------|----------|-------------|---|----------|---|-------------|-----------|-----------|----|------------|-----------|-----------|----|
| Ім'я змінної | «вудочка» | «погода» | «вудочка» | і | «погода» | і | «риболовля» | «наживка» | «наживка» | і | «клювання» | «рибалка» | «рибалка» | і |
| | 1-е правило | | 2-е правило | | ... | | | | | | | | | |

Таблиця 1.3

| № правила | № умови в правилі | Значення істинності |
|-----------|-------------------|---------------------|
| | | |

Таблиця 1.4

| Змінна | Ознака ініціалізації | Значення істинності |
|-----------|----------------------|---------------------|
| вудочка | NI | — |
| погода | NI | — |
| риболовля | NI | — |
| наживка | NI | — |
| черв'яки | NI | — |
| клювання | NI | — |
| рибалка | NI | — |
| помічник | NI | — |

1.2.5. Узагальнений алгоритм роботи експертної системи продукційного типу, яка реалізує пряме виведення

В експертних системах із *прямим виведенням* за відомими фактами відшукується висновок, який з цих фактів випливає. Якщо такий висновок вдається знайти, то він заноситься в робочу пам'ять. Тому пряме виведення називають виведенням, яке керується даними. Нижче наводиться узагальнений алгоритм роботи експертної системи прямого виведення, у якій рішення відшукується методом *пошуку завширки*:

1) Ініціювати роботу експертної системи шляхом ініціалізації змінних і присвоєння їм вихідних значень.

2) Занести змінні, які були ініційовані, в чергу змінних логічного виведення.

3) Переглянути список змінних умови та знайти змінну, ім'я якої знаходиться на початку черги змінних логічного виведення. Якщо змінна знайдена, записати в покажчик змінних умови номер відповідного правила. В протилежному випадку перейти до п.6.

4) Присвоїти значення неініційованим змінним умовної частини знайденого правила (якщо такі є) – для цього сформулювати відповідні запити користувачеві (імена змінних знаходяться в списку змінних умови). Перевірити значення істинності всіх умов правила та, якщо умовна частина є «істиною», перейти до п.5. У протилежному випадку перейти до п.3 і продовжити пошук.

5) Змінити стан робочої пам'яті шляхом ініціювання і присвоєння відповідного значення змінній, яка знаходиться в заключній частині поточного правила; помістити змінну в кінець черги змінних логічного виведення та помітити правило як таке, що спрацювало.

6) Якщо поточна змінна більше не зустрічається в умовній частині якого-небудь правила, видалити її з черги змінних логічного виведення. В протилежному випадку перейти до п.3.

7) Закінчити процес експертизи, як тільки спорожніє черга змінних логічного виведення. Якщо черга не порожня, повернутися до п.3.

8) Вивести користувачеві висновок експертної системи, що являє собою заключну частину останнього правила, яке спрацювало.

Пояснимо роботу експертної системи прямого виведення на прикладі. Припустимо, користувач описав поточну ситуацію за допомогою одного факту: «наживка = є». Цей факт заноситься в робочу пам'ять експертної системи і змінна «наживка» ставиться в чергу змінних логічного виведення. Потім у базі знань відшукується правило, умовна частина якого містить змінну «наживка». Для цього переглядається список змінних умови, восьмим елементом якого є «наживка». Таким чином впливає, що у покажчик змінних умови записується: № правила – 4; № умови – 2. Значення змінної «наживка» у робочій пам'яті й в умові № 2 правила № 4 збігаються, впливає, результат перевірки – «істина». Тому що сьомий елемент списку змінних умови містить змінну «риболовля», для активації правила № 4 необхідно перевірити її значення. Перевірка стану робочої пам'яті показує, що змінна «риболовля» не є ініційованою. У результаті користувачеві формується запит: «Повідомте, риболовля відбудеться чи ні?» Припустимо, користувач відповів, що «Риболовля = відбудеться». Цей факт міститься в робочу пам'ять, а змінна «риболовля» ставиться в чергу змінних логічного виведення після змінної «наживка» (рис. 1.3). У покажчик змінних умови записується: № правила – 4; № умови – 1. Значення змінної «риболовля» у робочій пам'яті й в умові № 1 правила № 4 збігаються, таким чином, результат перевірки – «істина». Оскільки умовна частина правила № 4 є «істиною», впливає, правило спрацює. У результаті змінна «кловання» у робочій пам'яті ініціюється, їй присвоюється значення «буде», і вона заноситься в чергу. На цьому цикл роботи механізму логічного виведення завершений.

У черговому циклі поточною змінною в черзі залишається «наживка». Механізм логічного виведення відшукує правило № 5 і записує його в покажчик. Значення змінної «наживка» у робочій пам'яті й в умові № 1 правила № 5 не збігаються, тому правило № 5 не активується і результат

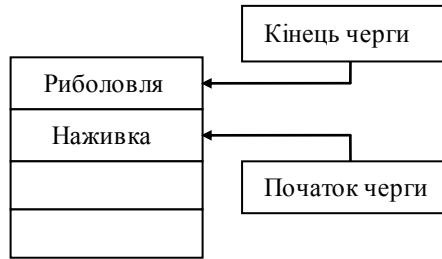


Рисунок 1.3 – Черга змінних логічного виведення

перевірки – «неправда» – записується в покажчик. Більше змінна «наживка» у списку змінних умови не зустрічається, вона видаляється з черги, і на цьому черговий цикл завершений.

У наступному циклі поточною змінною є «риболовля», вона у списку змінних умови не зустрічається, і тому видаляється з черги.

Тепер поточною змінною є «клювання», у покажчик записується: № правила – 6, № умови – 1, результат перевірки – «істина». Далі системі потрібно перевірити значення змінної «рибалка», що не була ініційована. Формується відповідний запит користувачеві, на який, припустимо, він відповів «рибалка = не досвідчений». Цей факт також заноситься в робочу пам'ять, а змінна «рибалка» – у чергу. Результат перевірки умови № 2 правила № 6 – «неправда», змінна «клювання» у списку більше не зустрічається, і тому видаляється з черги.

У черговому циклі поточною змінною є «рибалка», тому перевіряється умова № 1 правила № 7. Результат перевірки – «істина», правило спрацьовує, у робочу пам'ять заноситься факт «помічник = потрібний», а змінна «помічник» ставиться в чергу.

У наступному циклі змінна «рибалка» видаляється з черги. Потім з черги видаляється змінна «помічник». Оскільки черга змінних логічного виведення виявилася порожньою, процес експертизи закінчений і, як результат, користувачеві повідомляється висновок останнього правила, яке спрацювало: «Помічник потрібний».

Таким чином, використовуючи описані структури даних і алгоритм прямого виведення, експертна система здатна імітувати розумовий процес і робити осмислені висновки.

1.3. Постановка завдання

1) Розробити базу знань продукційного типу, яка містить 6 – 10 правил, відбиваючих причинно-наслідкові відношення між поняттями предметної області, в якій Ви вважаєте себе фахівцем.

2) Розробити структури даних, які потрібні для роботи механізму логічного виведення, що реалізує пряме виведення, з урахуванням створеної бази знань.

3) Розробити інтерфейс експертної системи, що дозволяє користувачеві:

а) вводити вихідну інформацію, яка ініціює роботу експертної системи;

б) одержувати остаточний результат роботи експертної системи;

в) за вимогою користувача одержувати ланцюг логічних висновків, який побудувала експертна система в процесі експертизи.

4) Написати програму експертної системи, що реалізує пряме виведення і пошук рішення методом *перебору за вищирішки* (програмне середовище вибрати самостійно).

5) Перевірити висновки експертної системи на тестових прикладах.

Лабораторна робота 2

**Розробка елементів експертної системи продукційного типу,
яка реалізує зворотне виведення**

2.1. Мета роботи

Вивчити склад і принципи роботи механізму логічного виведення експертної системи продукційного типу, яка реалізує зворотне виведення.

2.2. Інформаційний матеріал

2.2.1. Побудова дерева рішень

Існує клас задач, для яких простір пошуку рішення потенційно обмежено. До даного класу належать задачі, в процесі розв'язання яких необхідно вибрати один з декількох можливих варіантів рішення, який найбільш підходить за даним набором ознак. Наприклад, задача вибору посади для людини, яка бажає влаштуватися на роботу, залежно від її віку, кваліфікації, досвіду роботи, потреб фірми у фахівцях і т.д.

Для розв'язання задач такого класу зручно користуватися зворотним ланцюгом міркувань. Тому доцільною вважається розробка експертної системи, яка реалізує зворотне виведення.

Для опису процесу розв'язання подібних задач звичайно використовуються діаграми, які називаються *деревами рішень*. Вони наочно дозволяють простежити хід міркувань. Круги, що містять назви ознак, називаються *вершинами рішень*. Гілки дерева рішень закінчуються прямокутниками, що містять логічні висновки. Можна сказати, що вершини містять змінні, а шляхи – це умови, відповідно до яких змінним присвоюються значення.

На рис. 2.1 зображений приклад дерева рішень для розглянутої задачі.

2.2.2. Перетворення дерева рішень у базу знань продукційного типу

Для перетворення побудованого дерева рішень у базу знань продукційного типу використовується такий алгоритм:

- 1) Вибрати з дерева рішень вершину висновку й зафіксувати її.
- 2) Знайти пов'язану з нею вершину рішення й зафіксувати її.
- 3) Повторювати п.2 доти, поки не будуть вичерпані всі вершини рішення, пов'язані із зафіксованою вершиною висновку, або не зустрінеться нова вершина висновку.



Рисунок 2.1 - Дерево рішень

4) Кожна вершина рішення, яка складає шлях, – це одна зі змінних частини ЯКЩО правила. Змінні умовної частини правила поєднуються за допомогою логічної операції кон'юнкції.

5) Зафіксований логічний висновок перенести в частину ТО правила. Якщо дерево рішень ще не вичерпано, перейти до п. 1.

Для розглянутого прикладу база знань буде складатися із чотирьох правил:

1. ЯКЩО «вік ≥ 45 років», ТО «посада = сторож».
2. ЯКЩО «вік < 45 років»
І «рівень освіти = середня», ТО «посада = лаборант».
3. ЯКЩО «вік < 45 років»
І «рівень освіти = вища»
І «вид освіти = технічна», ТО «посада = програміст».
4. ЯКЩО «вік < 45 років»
І «рівень освіти = вища»
І «вид освіти = гуманітарна», ТО «посада = менеджер».

2.2.3. Структури даних, потрібні для реалізації механізму зворотного виведення

Типова експертна система продукційного типу, яка реалізує зворотне виведення, містить такі структури даних:

- 1) база знань – набір продукційних правил;

2) список змінних умови, тобто перелік змінних, які увійшли в умовну частину кожного правила. Являє собою масив, у якому для кожного правила резервується деяка кількість елементів;

3) стек змінних логічного висновку, у якому прийнятий принцип обробки: «останнім прийшов – першого обслужили»;

4) робоча пам'ять – містить значення змінних й ознаку їхньої ініціалізації. До початку експертизи ознака ініціалізації дорівнює NI (англ.: No Initialize) і всім змінним присвоюються порожні значення. У процесі експертизи змінним присвоюються поточні значення, а значення ознаки змінюється на I (англ.: Initialize);

5) показчик змінних умови – виконує роль протоколу роботи експертної системи. Являє собою масив, у який заноситься номер поточного правила, номер умови в правилі, що перевіряється в цей момент, і результат перевірки істинності;

6) список змінних логічного висновку - структура, що містить упорядкований список всіх існуючих логічних висновків.

2.2.4. Узагальнений алгоритм роботи експертної системи, яка реалізує зворотне виведення

1) Визначити змінну логічного висновку, яка цікавить користувача, і помістити її в стек.

2) У списку змінних логічного висновку шукати ім'я поточної змінної, яка міститься у верхівці стека. Якщо змінна знайдена, у показчик змінних умови записати номер відповідного правила й установити номер умови рівним 1. Якщо змінна не знайдена – виводиться повідомлення користувачеві.

3) Перевірити значення всіх змінних умови поточного правила.

4) Якщо змінна умови ініційована – перейти до п. 7.

5) Якщо в робочій пам'яті зазначено, що змінній умови не присвоєне значення, шукати її в списку змінних логічного висновку. Якщо змінна не знайдена – сформулювати відповідний запит користувачеві, присвоїти значення змінній і перейти до п.7. Якщо поточна змінна знайдена в списку змінних логічного висновку, помістити її в стек, а в показчик змінних умови записати номер відповідного правила, установити номер умови рівним 1 і перейти до п.3.

6) Порівняти значення змінної умови з робочої пам'яті зі значенням в умовній частині поточного правила. Якщо результат порівняння «істина» та якщо перевірені ще не всі змінні умови поточного правила, збільшити номер умови на 1 і перейти до п.4. Якщо результат порівняння «неправда», відзначити в протоколі, що правило не спрацювало і перейти до п.2.

7) Якщо перевірені всі змінні умови поточного правила і умовна частина має значення «істина», присвоїти значення змінній, яка знаходиться в заключній частині поточного правила, занести її в робочу пам'ять і видалити зі стека.

8) Якщо стек порожній – закінчити процес експертизи і видати користувачеві результат. У протилежному випадку – перейти до п.2.

2.3. Постановка завдання

1) Нарисувати дерево рішень для розв'язання інтелектуальних задач у тій предметній області, у якій Ви вважаєте себе фахівцем.

2) Розробити базу знань продукційного типу, використовуючи отримане дерево рішень.

3) Розробити структури даних, які потрібні для створення механізму логічного виведення зворотного типу, з урахуванням створеної бази знань.

4) Розробити інтерфейс експертної системи, що дозволяє користувачеві:

а) вводити вихідну інформацію, яка ініціює роботу експертної системи;

б) одержувати остаточний висновок системи, який є відповіддю на запит користувача;

в) на вимогу користувача одержувати логічний ланцюг, вибудований системою в процесі експертизи.

5) Написати програму експертної системи, яка реалізує зворотне виведення і пошук рішення методом *перебору завширшки* (програме середовище вибрати самостійно).

6) Перевірити висновки експертної системи на тестових прикладах.

Лабораторна робота 3
**Побудова механізму логічного виведення,
який здатен змінювати напрямок пошуку**

3.1. Мета роботи

Вивчити склад і принципи роботи механізму логічного виведення експертної системи, який в умовах неповноти даних здатен змінювати напрямок пошуку.

3.2. Інформаційний матеріал

Для того щоб експертна система могла ефективно вирішувати реальні практичні задачі, механізм логічного виведення (МЛВ) повинен уміти самостійно вибирати й змінювати напрямок висновку.

Для цього МЛВ повинен одночасно містити чергу змінних логічного виведення, що працює за принципом «першим прийшов – першого обслужили», і стек змінних логічного виведення, що працює за принципом «останнім прийшов – першого обслужили».

Для того щоб ініціювати роботу експертної системи, користувач повинен повідомити наявні факти, тобто ініціювати змінні робочої пам'яті, значення яких відомі. Ці змінні заносяться в чергу і включається механізм прямого виведення.

Якщо під час перевірки істинності умовної частини якого-небудь правила МЛВ знаходить неініційовану змінну, ця змінна заносяться в стек і включається механізм зворотного виведення. Зворотнє виведення працює доти, поки стек не буде очищений, тобто всім змінним, які були поміщеними у стек, не будуть присвоєні значення або в результаті спрацьовування правил, або за допомогою запитів користувачеві. Після видалення зі стека останньої змінної виведення знову перемикається на пряме і МЛВ продовжує роботу, поки черга змінних логічного виведення не виявиться порожньою.

Як результат експертизи користувачеві повідомляється останній зроблений системою висновок (висновок останнього правила, що спрацювало).

Наприклад, розглянемо дерево рішень, яке зображене на рис. 3.1. Для розглянутого приклада база знань буде складатися із трьох правил:

1. ЯКЩО «A = істина», ТО «B = істина».
2. ЯКЩО «C = істина», ТО «D = істина».
3. ЯКЩО «B = істина» І «D = істина», ТО «E = істина».

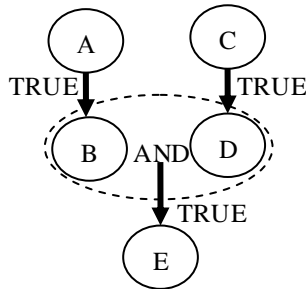


Рисунок 3.1 – Приклад дерева рішень

Нехай користувач повідомив, що є факт «A = істина». Тоді змінна A заноситься в робочу пам'ять й у чергу змінних логічного виведення. Оскільки черга не порожня, включається механізм прямого виведення. Відшукується Правило 1, установлюється, що його умовна частина складається з одного твердження і є істиною. Правило 1 спрацьовує, у результаті значення змінної B заноситься в робочу пам'ять, а ім'я змінної B – в чергу. Більше змінна A в умовних частинах правил, що перебувають у базі знань, не зустрічається, тому змінна A із черги видаляється. Поточною стає змінна B, механізм прямого виведення відшукує Правило 3 і встановлює, що для перевірки істинності його умовної частини необхідно встановити значення змінної D, яка є неініційованою. Змінна D заноситься в стек, і включається механізм зворотного виведення. Відшукується Правило 2 і встановлюється, що для перевірки істинності його умовної частини необхідно встановити значення змінної C, яка є неініційованою. Змінна C заноситься в стек, і потім механізм зворотного виведення не виявляє в базі знань правил, за допомогою яких можна встановити значення змінної C. Тоді механізм зворотного виведення формує запит користувачеві із проханням повідомити значення змінної C. Після одержання відповіді користувача значення змінної C заноситься в робочу пам'ять, вона видаляється зі стека й поточною стає змінна D, що перебуває у вершині стека. Механізм зворотного виведення перевіряє значення істинності умовної частини Правила 2, і, якщо вона є істиною, правило спрацьовує, значення змінної D заноситься в

робочу пам'ять і вона видаляється зі стека. Оскільки стек порожній, знову включається механізм прямого виведення, поточною стає змінна В, що перебуває в черзі, і перевіряється умовна частина Правила 3. Якщо вона є істиною, Правило 3 спрацює, значення змінної Е заноситься в робочу пам'ять, а її ім'я – в чергу. Більше змінна В в умовних частинах правил, що перебувають у базі знань, не зустрічається, тому вона із черги видаляється. Поточною стає змінна Е, що в умовних частинах правил, які перебувають у базі знань, не зустрічається, тому вона із черги видаляється. Черга стає порожньою, робота механізму логічного виведення припиняється і, як результат експертизи, користувачеві виводиться повідомлення «Е = істина».

3.3. Постановка завдання

1) Нарисувати дерево рішень для розв'язання інтелектуальних задач у предметній області, в якій Ви вважаєте себе фахівцем.

2) Розробити базу знань продукційного типу, використовуючи отримане дерево рішень.

3) Розробити структури даних, які потрібні для створення механізму логічного виведення, здатного змінювати напрямок пошуку, з урахуванням створеної бази знань.

4) Розробити інтерфейс експертної системи, який дозволяє користувачеві:

а) вводити вихідну інформацію, яка ініціює роботу системи;

б) одержувати остаточний висновок експертної системи, який подає заключну частину останнього правила, що спрацювало;

в) на вимогу користувача одержувати логічний ланцюг, вибудований системою в процесі експертизи.

5) Написати програму, яка реалізує механізм логічного виведення, що здатний змінювати напрямок пошуку (програмне середовище вибрати самостійно).

6) Налаштувати роботу механізму логічного виведення на тестових прикладах.

Лабораторна робота 4

**Розробка механізму обчислення коефіцієнтів упевненості
для експертної системи прямого виведення**

4.1. Мета роботи

Вивчити принципи обчислення коефіцієнтів упевненості продукційних правил, якщо відомі коефіцієнти впевненості умовних частин правил. Побудувати експертну систему, яка використовує методику коефіцієнтів упевненості.

4.2. Інформаційний матеріал

Метод коефіцієнтів упевненості (КУ) був уперше застосований в експертній системі MYCIN [2]. Він базується на евристичних міркуваннях, які були запозичені із практичного досвіду роботи експертів. Коли експерт оцінює ступінь вірогідності деякого висновку, він використовує такі поняття, як «точно», «досить ймовірно», «можливо» і т.п. Розроблювачі системи MYCIN вирішили відобразити ці розмиті поняття на шкалу коефіцієнтів упевненості, які змінюються в діапазоні від -1 до $+1$. Якщо коефіцієнт упевненості набуває значення, яке дорівнює $+1$, то факт вважається «істиною». Якщо -1 , то факт є «неправдою». Шкала зміни значень коефіцієнтів упевненості зображена на рис. 4.1.

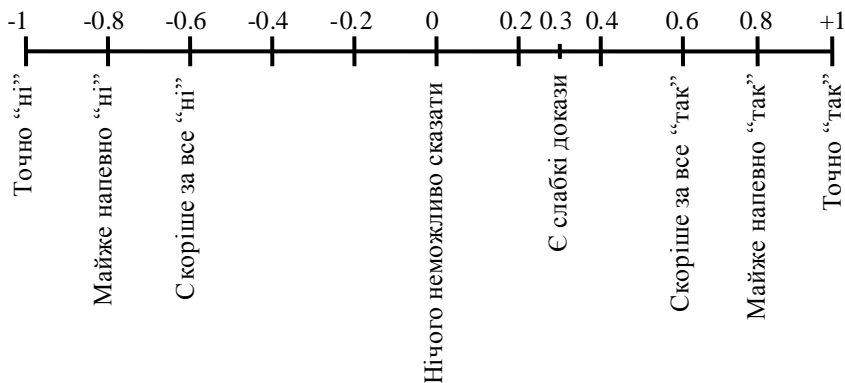


Рисунок 4.1 – Шкала коефіцієнтів упевненості

Оскільки евристичні правила продукційного типу ґрунтуються винятково на людському досвіді, з повною визначеністю ніколи не можна сказати, що вони правильні. Ось чому коефіцієнти упевненості приписуються не тільки фактам, але й правилам. Таким способом забезпечується облік ненадійності правил.

У ході логічного виведення над фактами, що становлять передумови продукційних правил, виконуються логічні операції. У результаті цього утворюються складні висловлення.

Загальний алгоритм обчислення коефіцієнта упевненості висновку правила у випадку, коли фактам умовної частини правила присвоєні деякі коефіцієнти упевненості:

1) Вибрати мінімальне значення КУ для висловлень умовної частини правила, які зв'язані логічним оператором «І».

2) Вибрати максимальне значення КУ для висловлень умовної частини правила, які зв'язані логічним оператором «АБО».

3) Помножити обраний КУ на КУ правила, який експерт приписав даному правилу.

4) У процесі логічного виведення той самий висновок може підтверджуватися різними правилами, кожне з яких приписує висновку свій КУ. Очевидно, що коефіцієнт упевненості висновку, підтверджуваного декількома правилами, повинен збільшуватися. Чим більше є підтверджень на користь деякого висновку, тим ближче до одиниці повинен бути його КУ. Ось чому, якщо знайдено два правила з однаковим логічним висновком, потрібно перерахувати КУ висновку у відповідності до формули:

$$КУ_{\text{Новий}} = КУ_{\text{Раніше обчислений}} + (1 - КУ_{\text{Раніше обчислений}}) \times КУ_{\text{Нового висновку}}$$

У багатьох випадках експерт при формулюванні правил задає граничне значення коефіцієнта упевненості правила. Логічний висновок вважається правильним тільки в тому випадку, якщо його коефіцієнт упевненості перевищує задане граничне значення. Робота механізму логічного виведення триває доти, поки не спрацює правило, у якому коефіцієнт упевненості висновку виявиться більшим, ніж встановлене експертом граничне значення.

4.3. Постановка завдання

1) Відредагувати базу знань, розроблену в лабораторній роботі 1.

Для цього:

а) присвоїти усім правилам деякі значення коефіцієнтів упевненості (наприклад, 0,9);

б) задати граничні значення коефіцієнтів упевненості для кожного правила (наприклад, 0,5).

2) Відредагувати інтерфейс, розроблений у лабораторній роботі 1, так, щоб користувач мав можливість ввести числове значення коефіцієнта упевненості, яке визначає ступінь його впевненості у вірогідності відповіді.

3) Ввести в механізм логічного виведення, розроблений у лабораторній роботі 1, блок обчислення коефіцієнтів упевненості правил, виходячи із заданих користувачем коефіцієнтів упевненості умовних частин правил.

4) Відредагувати програму, розроблену в лабораторній роботі 1, так, щоб продукційне правило спрацьовувало тільки в тому випадку, коли коефіцієнт упевненості, який був обчислений для висновку правила, виявився більшим, ніж задане граничне значення.

Лабораторна робота 5

Підвищення ефективності висновків механізму логічного виведення зворотного типу за допомогою методики Байєса

5.1. Мета роботи

Ознайомитися з принципами використання байєсового підходу для побудови логічного виведення в умовах невизначеності. Розробити механізм логічного виведення зворотного типу на основі байєсового підходу.

5.2. Інформаційний матеріал

У багатьох випадках механізму логічного виведення (МЛВ) експертної системи потрібно здійснювати висновок в умовах, коли вихідні

дані не є абсолютно точними й достовірними. Ця невизначеність може мати всіляку природу: може породжуватися неповнотою опису ситуації, ймовірносним характером спостережуваних подій, неточністю подання даних та ін. Основним поняттям, яке використовується при побудові моделей виведення в умовах невизначеності, є поняття вірогідності. Вірогідність висновків може бути визначена на основі різних підходів, з яких найбільш часто використовується ймовірнісна методика Байєса.

При байєсовому підході ступінь вірогідності кожного з фактів оцінюється ймовірністю, яка набуває значення в діапазоні від 0 до 1. Ймовірності вихідних фактів визначають або методом статистичних випробувань, або опитуванням експертів. Ймовірності висновків визначають на основі правила Байєса для обчислення апостеріорної умовної ймовірності $p(H|E)$ події (гіпотези) H за умови, що відбулася подія (свідчення) E :

$$p(H|E) = \frac{p(E, H)}{p(E)}, \quad (5.1)$$

де $p(E)$ – безумовна (апріорна) ймовірність свідчення E ; $p(E, H)$ – спільна ймовірність свідчення E і гіпотези H . Спільна ймовірність $p(E, H)$ дорівнює добутку безумовної ймовірності гіпотези $p(H)$ на умовну ймовірність того, що свідчення (факт) E має місце, якщо спостерігається гіпотеза H :

$$p(E, H) = p(H) \cdot p(E|H). \quad (5.2)$$

Звідси випливає, що апостеріорну ймовірність $p(H|E)$ можна обчислити за допомогою формули

$$p(H|E) = \frac{p(H) \cdot p(E|H)}{p(E)}. \quad (5.3)$$

Однак практичне застосування формули (5.3) викликає труднощі, оскільки значення апріорних ймовірностей $p(H)$ і $p(E)$, а також умовної ймовірності $p(E|H)$ можна з'ясувати тільки в експертів на етапі формування бази знань.

Якщо розглядається повна група несумісних гіпотез H_1, H_2, \dots, H_n з апріорними ймовірностями $p(H_1), p(H_2), \dots, p(H_n)$, то апостеріорну

ймовірність кожної з гіпотез при реалізації свідчення E обчислюють за допомогою формули, яку називають теоремою гіпотез Байєса:

$$p(H_i|E) = \frac{p(H_i) \cdot p(E|H_i)}{\sum_{k=1}^n p(E|H_k) \cdot p(H_k)} . \quad (5.4)$$

У розглянутих формулах передбачається, що свідчення E повністю визначене, тобто підтверджується з ймовірністю 1. Однак на практиці факти можуть підтверджуватися з меншою ймовірністю. Це може відбуватися в результаті логічного виведення, коли висновки одних правил виступають у ролі фактів (свідчень) інших правил. У цьому випадку врахувати ненадійність фактів при обчисленні ймовірності гіпотези дозволяє наступна формула:

$$p(H) = p(H|E) \cdot p(E) + p(H|\bar{E}) \cdot p(\bar{E}) . \quad (5.5)$$

Правила логічного виведення допускають об'єднання свідчень за допомогою логічних операцій І, АБО, НІ. У такій ситуації обробка кожного правила виконується з урахуванням принципів нечіткої логіки, розглянутих у розділі 4.2. Для фактів (свідчень), зв'язаних логічною операцією І, вибирається мінімальна з імовірностей. Для логічної операції АБО - вибирається максимальна з імовірностей. Операція логічного заперечення НІ приводить до обчислення зворотної ймовірності.

При використанні байєсового підходу виникають дві основні проблеми. По-перше, повинні бути відомі всі апіорні умовні ймовірності свідчень, а також апіорні ймовірності гіпотез. По-друге, з умов теореми Байєса випливає, що всі гіпотези, які розглядає експертна система, повинні бути несумісними. Проте даний підхід широко застосовується в експертних системах завдяки гарному теоретичному обґрунтуванню.

Розглянемо застосування байєсового підходу на прикладі наступного набору продукційних правил:

1. ЯКЩО «процентні ставки = падають»,
ТО «рівень цін на біржі = зростає».
2. ЯКЩО «процентні ставки = зростають»,
ТО «рівень цін на біржі = падає».
3. ЯКЩО «курс долара = падає», ТО «процентні ставки = зростають».
4. ЯКЩО «курс долара = зростає», ТО «процентні ставки = падають».

Крім правил, у базі знань повинні зберігатися значення апріорних умовних імовірностей, які отримані або методом статистичних випробувань, або в результаті опитування експертів:

$$p(\text{«процентні ставки = падають»} \mid \text{«курс долара = зростає»}) = 0,8;$$

$$p(\text{«процентні ставки = падають»} \mid \text{«курс долара = не зростає»}) = 0,1;$$

$$p(\text{«рівень цін на біржі = зростає»} \mid \text{«процентні ставки = падають»}) = 0,85;$$

$$p(\text{«рівень цін на біржі = зростає»} \mid \text{«процентні ставки = не падають»}) = 0,1.$$

Задача: визначити ймовірність підвищення рівня цін на біржі.

Для розв'язання задачі підходить механізм логічного виведення зворотного типу. Його робота почнеться з того, що гіпотеза «рівень цін = зростає» буде занесеною в стек, після чого в базі знань у заключних частинах правил МЛВ буде шукати висновок «рівень цін = зростає». Таке правило існує. Перепишемо правило 1 відповідно до формули (5.5):

$$1. p(\text{«рівень цін = зростає»}) = p(\text{«рівень цін = зростає»} \mid \text{«ставки = падають»}) \times \\ \times p(\text{«ставки = падають»}) + p(\text{«рівень цін = зростає»} \mid \text{«ставки = не падають»}) \times \\ \times p(\text{«ставки = не падають»}).$$

Тепер новою метою стає гіпотеза «процентні ставки = падають», яка заноситься в стек і відшукується у заключних частинах правил у базі знань.

Це – правило 4. Перепишемо його відповідно до формули (5.5):

$$4. p(\text{«ставки = падають»}) = p(\text{«ставки = падають»} \mid \text{«курс = зростає»}) \times \\ \times p(\text{«курс = зростає»}) + p(\text{«ставки = падають»} \mid \text{«курс = не зростає»}) \times \\ \times p(\text{«курс = не зростає»}).$$

Новою метою стає гіпотеза «курс долара = зростає», яка теж заноситься в стек, але в заключних частинах правил не зустрічається. Тому система запрошує значення ймовірності цього факту в користувача. Нехай, наприклад, користувач відповів, що $p(\text{«курс = зростає»}) = 0,9$. Після відповіді користувача ця гіпотеза зі стека видаляється. Тоді МЛВ зможе обчислити:

$$p(\text{«курс = не зростає»}) = 1 - p(\text{«курс = зростає»}) = 1 - 0,9 = 0,1.$$

Підставимо отримані значення в правило 4:

$$p(\text{«ставки = падають»}) = 0,8 \times 0,9 + 0,1 \times 0,1 = 0,73.$$

Поточна гіпотеза є правильною, вона видаляється зі стека, крім того, МЛВ може обчислити:

$$p(\text{«ставки = не падають»}) = 1 - p(\text{«ставки = падають»}) = 1 - 0,73 = 0,27.$$

Підставимо отримані значення в правило 1:

$$p(\text{«рівень цін = зростає»}) = 0,85 \times 0,73 + 0,1 \times 0,27 = 0,65.$$

Поточна гіпотеза є правильною, вона видаляється зі стека, після чого він залишається пустим. Робота механізму логічного виведення зупиняється, і користувач отримує відповідь експертної системи: «Ймовірність підвищення рівня цін на біржі дорівнює 65 %».

5.3. Постановка завдання

1) Відредагувати базу знань, розроблену в лабораторній роботі 2. Для цього потрібно для кожного правила визначити значення апріорних умовних ймовірностей свідчення та висловлення, яке є зворотним.

2) Відредагувати інтерфейс, розроблений у лабораторній роботі 2, так, щоб користувач мав можливість ввести числове значення ймовірності, яке визначає вірогідність його відповіді.

3) Ввести в механізм логічного виведення зворотного типу, який був розроблений у лабораторній роботі 2, блок обчислення ймовірності висновку на основі байєсового підходу.

4) Перевірити роботу механізму логічного виведення на тестових прикладах.

СПИСОК ЛІТЕРАТУРИ

1. **Вступ до експертних систем** : навч. посібник / В.О. Кравець, І.П. Хавіна, Ю.М. Колибін та ін.; НТУ "ХПІ". – Х.: НТУ "ХПІ", 2006. – 232 с.

2. **Джарратано Дж., Райли Г.** Экспертные системы: принципы разработки и программирование, 4-е издание. : Пер. с англ. – М. : ООО “Издательский дом Вильямс”, 2007. – 1152 с.

3. **Джексон П.** Введение в экспертные системы : пер. с англ. : учеб. пособие. – М. : Издательский дом “Вильямс”, 2001. – 624 с.

4. **Уотермен Д.** Руководство по экспертным системам : пер. с англ. – М.: Мир, 1989. – 388 с.

Зміст

| | |
|---|----|
| ВСТУП | 3 |
| Лабораторна робота 1 Розробка елементів експертної системи продукційного типу, яка реалізує пряме виведення | 4 |
| Лабораторна робота 2 Розробка елементів експертної системи продукційного типу, яка реалізує зворотнє виведення | 13 |
| Лабораторна робота 3 Побудова механізму логічного виведення, який здатен змінювати напрямок пошуку | 17 |
| Лабораторна робота 4 Розробка механізму обчислення коефіцієнтів упевненості для експертної системи прямого виведення | 20 |
| Лабораторна робота 5 Підвищення ефективності висновків механізму логічного виведення зворотного типу за допомогою методики Байєса | 22 |
| СПИСОК ЛІТЕРАТУРИ | 26 |

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних занять з дисципліни
“ЕКСПЕРТНІ СИСТЕМИ ТА БАЗИ ЗНАНЬ”
для студентів спеціальностей
7.080201 “Інформатика”,
7.080202 “Прикладна математика”,
7.080203 “Системний аналіз і управління”

Укладач ДОРОФЄЄВ Юрій Іванович

Відповідальний за випуск О.С. Куценко

Роботу до видання рекомендував О.В. Горєлий

Редактор О.С. Самініна

План 2009 р., поз. 6/67-09

Підп. до друку 06.05.2009. Формат 60×84 1/16. Папір офсетний.
Друк – ризографія. Гарнітура Таймс New Roman. Ум. друк. арк. 1,6.
Наклад 50 прим. Зам. № 138. Ціна договірна.

Видавничий центр НТУ “ХП”.
Свідоцтво про державну реєстрацію ДК №116 від 10.07.2000 р.
61002, Харків, вул.Фрунзе, 21

Друкарня НТУ “ХП”.
61002, Харків, вул.Фрунзе, 21